Fall 2019

CS 288-001: Intensive Programming in Linux

Xiaoning Ding

CS 288: Intensive Programming in Linux

Fall 2019

Tuesdays and Thursdays 11:30 am - 12:50 pm, Kupfrian Hall 117

Lecture Instructor: Xiaoning Ding
Phone: (973)596-3390
E-mail: xiaoning.ding@njit.edu
Office Hours: Tuesday: 4:00pm ~ 5:30pm, Thursday: 4:00 pm ~ 5:30pm, or by appointment
Office Location: GITC 4203
Teaching Assistant: TBA

Course Description

The course covers programming topics in Linux OS using C, Python, and PHP as primary languages. The course consists of the following modules: basic tools and utilities for system-level software development, such as Bash, C programming, and Unix/Linux system API; methods and algorithms for processing web data, such as searching trees and matrix computing; end-to-end applications such as one that constantly presents top 100 stocks; and extending the applications to run on multiple machines. The course provides students with utilities and hands-on experience for programming relatively large applications.

- This course is NOT an introduction to specific programming languages.
- This course uses intensively Linux and Linux command line interface (shell and command lines, NOT graphic user interface). But it is NOT an introduction to Unix/Linux systems or Unix/Linux command lines and tools.
- This course is NOT an introduction to operating systems.

Specific Goals for the Course

This course is intended for students who want to improve their programming skills and system skills in Linux. The goals of the course are to teach the programming facilities that are required to write robust programs at different levels of software systems and in different scales, to sharpen their programming skills by working on a variety of assignments and projects with these programming facilities, and to improve their understanding on the software development and software development ecosystem in Linux.

The Students Will Be Able to:

- Write shell scripts
- Use existing tools and develop new tools to process text contents
- Implement sorting and searching algorithms using pointers in C
- Develop system software using Linux system API and C
- Test programs
- Writing web processing programs
- Explain web services, describe web service stacks, and develop web applications

- Develop parallel/distributed programs that use multiple processes and run on multiple machines

Student Outcomes

- An intimate understanding of the Unix/Linux systems and programming environments;
- An ability to develop complex system-level software in the C programming language;
- An understanding of web service stacks and web development platform;
- An ability of developing applications on web development platforms;
- An ability of developing distributed/parallel applications;
- An ability of testing programs.

Prerequisites

- CS 114 Introduction to Computer Science II
- CS 280 Programming Language Concepts
- The course is about programming in Linux systems, and you will be using intensively Linux command line interface, NO GUI (graphic user interface). You need to know Linux systems and common Linux commands to understand course materials and finish assignments.
- The course uses intensively C language. You need to know how to read and write C programs to understand course materials and finish assignments.
- The course needs you to write runnable, robust, and relatively large applications. Some applications build and manage data structures using C pointers. It is important that you know how to debug a C program, e.g., how to use gdb.

Textbook

The C Programming Language, Kernighan and Ritchie, Prentice Hall, 2nd ed., ISBN: 978-0131103627

Note: We will NOT go over any materials in the textbook. But you probably need to refer to the book when you read or write C programs.

Recommended Books

Organization and Design, Patterson and Hennessy, Morgan Kaufmann, 4th ed., ISBN: 978-0123744937
Linux System Programming, Robert Love, O'Reilly Media, 2nd ed., ISBN: 978-1449339531

Online Materials

URLs will be posted on course home page in Canvas.

Course Work and Grading

Your grade in the course will be determined by the following breakdown:

- Quizzes – 5%

1~3 in-class quizzes on unannounced dates. 15-minute each.

- Programming Assignments – 10%
  There are two parts. **Minor assignments (5%)** require students to write small programs. **Major assignments (5%)** require students to write relatively large programs. Depending on the progress of course, there will be about 6 minor assignments, and 5 major assignments. Major assignments are 1) text searching using regex, 2) radix sort, 3) 15 puzzle with A* search, 4) traverse a directory and examine all the files, and 5) mini-web server.

  In minor assignments, students need to submit their programs and the proofs that can show the correct executions of the programs (e.g., screenshots of executions). Students get credits by submitting the required documents.

  In major assignments, students submit their own programs and test the programs submitted by others. Based on the tests, grades are assigned to programs for students' reference, but are not used when calculating the overall grades. Students get credits by submitting their own programs and finishing testing the programs of peer students. The credits are used when calculating the overall grades of the course.

  Develop your own programs. Do NOT copy or reengineer the programs that were not developed by you (e.g., those from other students in the class, previous classes of the course, internet, etc).

- Midterm tests – 30%
  A midterm will be scheduled in the 8th week of the semester. Most questions in midterm exam will be derived from examples and programming assignments.

- Final exam – 55%
  Final exam will be scheduled by the University. Check online for the time and location. Most questions in final exam will be derived from examples and programming assignments.

Grading Policies:

Based NJIT CS Grading Standard for 100- and 200- level courses:
A <= 15% of students
B+ <= 15% of students
B <= 15% of students
C+ <= 15% of students
C <= 15% of students
D/F/W remainder

Schedule

* subject to change according to class pace

Week	Topic
1	Overview (Introduction to Linux)
2, 3, 4	bash scripting, regular expressions, C pointers and binary operations

5, 6, 7	radix-sort, A* search, programs with multiple processes
8, 9, 10	Midterm, file operations, IPC
11, 12, 13	LAMP, HTTP, web processing, connecting to database servers, web applications
14, 15	parallel/distributed program development

Course Policies

- Use your NJIT email address to send emails and include "CS288" and the section number in the subject line. Your emails may be filtered out if you don't follow. When you receive emails from the course, read the whole emails, not just subject lines.
- Visit the course homepage in Canvas regularly and frequently for lecture notes, homework assignments, instructions, and latest updates.
- You may discuss the exercises in general terms with your classmates, but you must come up with your own solutions and all written work must be your own. Copying programs or written assignments is a violation of NJIT honor code.
- Policy for late submissions
  - You are given 3 days of "grace period" (self-granted extensions) in total, which you can use to give yourself extra time without penalty. You can split the grace period and apply to multiple assignments. To use your grace days, you must include a request as *a submission comment* (on the submission web page) when you submit your work in Canvas. Include in the request how many grace days you want to use. Grace period must be used in whole days. A grace period requested in hours will be rounded up to days (e.g., 11 hours will be rounded up to 1 day).
  - Instructor-granted extensions are only considered after all self-granted days are used and only given in exceptional situations (e.g., jury duty, medical problem).
  - Late work handed in when you have run out of grace is discounted 50% per day late.
  - Backup your work regularly. Losing files or accesses to any systems is not a valid excuse to late submission.
- Solutions to homework assignments will not be posted. The detailed requirements and the instruction for testing in each homework assignment will tell you if your programs are correct.
- Exams will be closed-book and closed-notes. But you are allowed to bring a letter-size cheat-sheet. Collaboration of any kind is not allowed in any exams. You are not allowed to take the exam of another section. Students with special needs are advised to make arrangements with Disability Services for exam accommodations.
- As a general rule, no makeup exams will be given, and no alternate dates for exams without a legitimate reason (e.g., jury duty, medical problem).
- The programming assignments and exams in this class are copyrighted, including the problem descriptions and programming hints that are posted/printed. Please do not post or

distribute problems and problem descriptions without written permissions from Prof. Xiaoning Ding. You may take notes and make copies of course materials for your own use. You may not and may not allow others to reproduce or distribute lecture notes and course materials publicly whether or not a fee is charged.

- NJIT policy on video recording class materials: You may not video record the class materials. You may not put any video/audio recorded class materials on the Web/Internet. It is against the University policy.
- Turn off cell phones during class unless you are otherwise instructed.

Honor Code

Academic Integrity is the cornerstone of higher education and is central to the ideals of this course and the university. Cheating is strictly prohibited and devalues the degree that you are working on. As a member of the NJIT community, it is your responsibility to protect your educational investment by knowing and following the academic code of integrity policy that is found at: http://www5.njit.edu/policies/sites/policies/files/academic-integrity-code.pdf.

Please note that it is my professional obligation and responsibility to report any academic misconduct to the Dean of Students Office. Any student found in violation of the code by cheating, plagiarizing or using any online software inappropriately will result in disciplinary action. This may include a failing grade of F, and/or suspension or dismissal from the university. If you have any questions about the code of Academic Integrity, please contact the Dean of Students Office at dos@njit.edu.